Murtaza, F., Yousaf, M.H. y Velastin, S.A. (2018). DA-VLAD: Discriminative Action Vector Of Locally Aggregated Descriptors for Action Recognition. In *2018 25th IEEE International Conference on Image Processing (ICIP)*.

DOI: https://doi.org/10.1109/ICIP.2018.8451255

DA-VLAD: DISCRIMINATIVE ACTION VECTOR OF LOCALLY AGGREGATED DESCRIPTORS FOR ACTION RECOGNITION

Fiza Murtaza[⋆] *Muhammad Haroon Yousaf*[⋆] *Sergio A. Velastin SMIEEE*[†]

[⋆] Department of Computer Engineering, University of Engineering and Technology Taxila, Pakistan
[†]University Carlos III Madrid, Spain and Queen Mary University of London, UK

ABSTRACT

In this paper, we propose a novel encoding method for the representation of human action videos, that we call Discriminative Action Vector of Locally Aggregated Descriptors (DA-VLAD). DA-VLAD is motivated by the fact that there are many unnecessary and overlapping frames that cause non-discriminative codewords during the training process. DA-VLAD deals with this issue by extracting class-specific clusters and learning the discriminative power of these codewords in the form of informative weights. We use these discriminative action weights with standard VLAD encoding as a contribution of each codeword. DA-VLAD reduces the inter-class similarity efficiently by diminishing the effect of common codewords among multiple action classes during the encoding process. We present the effectiveness of DA-VLAD on two challenging action recognition datasets: UCF101 and HMDB51, improving the state-of-the-art with accuracies of 95.1% and 80.1% respectively.

Index Terms— Human action recognition, VLAD, feature encoding, codewords, improved dense trajectories (iDT)

1. INTRODUCTION

Human action recognition has received significant attention from the computer vision community due to its large pool of applications including surveillance, automation, entertainment and several others. Action recognition is still a challenging task because of inconsistency in the temporal scale and periodicity in the human actions, the complex nature of motion, the exponential nature of all possible action categories and the dynamic background. The pipeline of human action recognition can be divided into three major steps: extraction of features from raw videos, encoding of the extracted features for proper video representation and the classification of this video representation into one of the predefined classes. Existing classification techniques are more developed, but feature extraction and encoding methods need improvements. In the literature there exist two types of feature extraction methods: hand-crafted and Convolutional Neural networks (CNNs) based features. Histograms-of-Oriented-Gradients (HOG) [1], Histograms-of-Optical-Flow (HOF) [2] and Motion-Boundary-Histograms (MBH) [3] are considered as the most popular hand-crafted feature descriptors. Different sampling techniques exist in the literature to extract regions of interest, on which these descriptors are applied, such as dense sampling [4] and motion trajectories [5, 6] etc. Recently, CNN-based representation shows impressive results for image classification [7]. For video classification, CNN-based representations [8–10] have not yet achieved significant success compared to the best hand-crafted feature descriptors [6]. One reason for this is that the current video datasets [11, 12] are rather small and contain only few thousands videos with few hundred action classes.

Fig. 1. Discriminative power of the frames taken from the HMDB51 dataset for action Hit (top) and Punch (bottom). The bar in the bottom shows the weights estimated during our proposed weighting scheme.

Similar to feature extraction, feature aggregation/encoding is an important task in a human action recognition framework. Different encoding methods such as improved Fisher Vectors (iFV) [13], VLAD [14], Spatio-Temporal Vector of Locally Max Pooled Features (ST-VLMPF) [15], Spatio-temporal VLAD (ST-VLAD) [16], ActionVLAD [17], AdaScan [18], MoFAP [19], Generalized rank pooling (GRP) [20], Modified-VLAD [21] have shown state-of-the-art performance in action recognition [22,23]. Similarly, Dual Adaptive VLAD (DuA-VLAD) is proposed in [24] which adopt the new cluster centers for the task of image retrieval. Another modified version of VLAD is Vectors of locally aggregated tensors (VLAT) [25] which sums the tensor product of the descriptors for the task of image classification. Both of these methods (DuA-VLAD and VLAT) are not tested for the task of human action recognition. These encoding methods have a few shortcomings that affect overall classification accuracy. First, they are built upon global clustering therefore they have no information about the class they belong to, hence how distinctive these codewords in the specific class are is unknown. Second, globally extracted codewords cannot model the inter-class variation effectively because features from different classes are most likely to be assigned to the same codeword during the encoding process.

As a solution to the aforementioned issue, we propose a novel encoding approach which generates more informative video representation. We propose a class-specific clustering approach for codebook creation by utilizing the available video-level class labels of the training data. We assign the weights to the codewords, based on their ability to discriminate among different action classes, i.e. high weights are assigned to the codewords that are best matched with the features of their own action classes. This issue is challenging because videos contain many frames that are common to other action classes, e.g. similar poses in hit and punch classes (as shown in Fig.

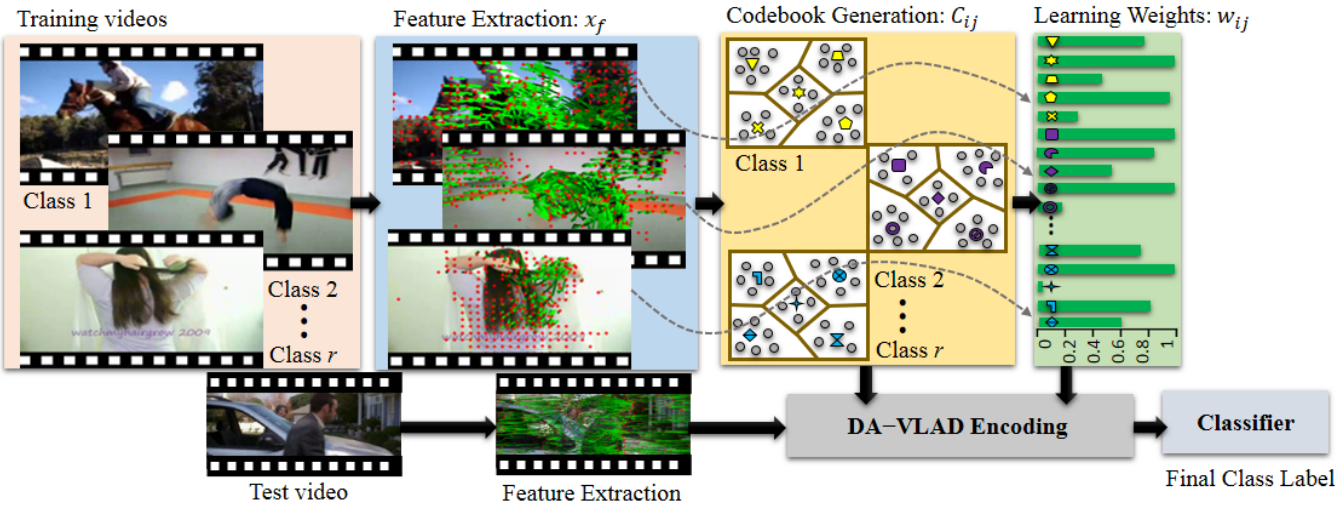

Fig. 2. Framework of the proposed approach

1) and also some background frames are common to multiple action classes.

The contributions of our proposed work are threefold: (1) First, we propose a novel codeword weighting scheme for ranking the relative importance of the codewords according to their discriminative power among different action classes. (ii) Second, we efficiently integrate the discriminative power of the codewords in the encoding process by taking into account the weights of the codewords (iii) Finally, we show that DA-VLAD outperforms other encoding schemes on two diverse action recognition datasets.

The rest of the paper is organized as follows: Section 2 formulates our encoding method. The experimental settings, results and comparisons are provided in Section 3. Finally, the conclusions are given in Section 4.

2. PROPOSED DA-VLAD ENCODING SCHEME

DA-VLAD seeks to model the inter-class variation efficiently to discriminate among different action categories. We present our proposed action recognition framework in Fig. 2. First, we extract and sample features from the videos of training data, and build a codebook of Discriminative action codewords by learning weights of each codeword (Section 2.1). These discriminative codewords are used to aggregate features from the training and testing videos into a fixed-length feature vector per video (Section 2.2). This representation will serve as an input to the classifier for the final classification task.

2.1. Learning Discriminative Action Codewords

To determine the Discriminative action codewords from the training data, we perform two successive steps. First, we obtain the codewords for each action class by performing clustering on the features set belonging to the corresponding action class, and we call it class-specific clustering. We prefer class-specific clustering over global clustering because we suppose that the features within a specific action class would not be independent but instead they share commonalities with the features of other classes. Therefore, features from different action classes are most likely to be assigned to the same codeword during the encoding process. We represent each video as a set of D-dimensional feature descriptors $x_f \in \mathbb{R}^D$, which are concatenated in matrix form as $V_m = [x_1|x_2|\cdots|x_{d_m}]$ with d_m equal to the total number of feature descriptors extracted from the m^{th} video. From the training set of r action classes $A = \{a_1, a_2, \cdots, a_r\}$, we compile all features of action a_i into a feature matrix $X_i \in \mathbb{R}^{D\times n_i}$ with n_i equal to the total number of features in the training set of action a_i. To perform class-specific clustering, we divide the feature matrix X_i into K action codewords using K-means clustering with Euclidean distance. In this way, for a total of r action classes we obtain $r \times K$ action codewords C_{ij}, where C_{ij} represents the j^{th} codeword of the action a_i.

Second, we compute the relative importance of the action codewords by obtaining their weights w_{ij} according to their ability to differentiate between different action classes. From training, feature descriptors $\{x_1, \cdots, x_n\}$ (with $n = n_i \times r$), each feature descriptor x_f (line 4 of Algorithm 1) is assigned to the nearest action codeword C_{ij} such that $||x_f - C_{ij}||$ is minimum. For each action codeword C_{ij}, we record the within-class q_{ij} and out-of-class q'_{ij} assignments (lines 5-9 of Algorithm 1) of the action codeword C_{ij} which corresponds to the correct and false assignments to C_{ij} respectively. For each action codeword C_{ij} we find its weight w_{ij} given by:

$$w_{ij} = \frac{q_{ij}}{q_{ij} + q'_{ij}} \qquad \forall\, i \in [1:r] \text{ and } j \in [1:K] \qquad (1)$$

From the viewpoint of action codewords; if an action codeword is common to many action classes it will have lower weight which will decrease its importance. From the viewpoint of action classes, action words which are assigned only to the feature descriptors of their corresponding action classes are discriminative therefore they have high weights. This procedure will also suppress the effect of those action codewords which correspond to the background regions of the frames of videos because these regions are common to most of the action classes.

2.2. DA-VLAD encoding

After calculating the action codewords C_{ij} and their corresponding weights w_{ij}, each video descriptor x_f from the set V_m =

Algorithm 1 Finding the number of within-class q_{ij} and out-of-class q'_{ij} assignments for $C_{ij}, \forall i \in [1:r]$ and $j \in [1:K]$

Input: Feature descriptors $\{x_1, \cdots, x_n\}$ and codewords C_{ij}
Output: q_{ij} and q'_{ij}

```
q_ij = 0 , q'_ij = 0
for all actions a_i ∈ A do
    for all features x_f ∈ a_i do
        Assign x_f to C_ij such that ||x_f − C_ij|| is minimum and find q_ij and q'_ij
        if C_ij ∈ a_i then
            q_ij = q_ij + 1
        else
            q'_ij = q'_ij + 1
        end if
    end for
end for
```

$[x_1|x_2|\cdots|x_{d_m}]$ is then assigned to its nearest action codeword. For densely sampled features, VLAD encoding performs better than iFV as the second-order statistics do not aid in obtaining higher accuracy, but add computational cost [15]. Similar to the standard VLAD, a residual vector $||x_f - C_{ij}||$ (which finds the difference between the feature descriptor and the assigned codeword) is computed for train and test videos. For each action codeword, we do the weighted average pooling over the residual vectors as:

$$v_{ij} = w_{ij} \times \frac{1}{N_{ij}} \sum_{f=1}^{Nij} (x_f - C_{ij}) \tag{2}$$

where N_{ij} represents the total number of feature descriptors assigned to the action codeword C_{ij}. The multiplication by w_{ij} results in discriminative representation therefore we call it as Discriminative Action VLAD (DA-VLAD) encoding. This multiplication with the corresponding weights will dominate or highlight the importance of the action codewords according to their capability to discriminate among different action classes. For the specific action video, all resulted vectors v_{ij} are concatenated into a single DA-VLAD encoded vector of size $r \times K \times D$. In Section 3.4, we discuss the effect of using only highly discriminating action codewords, from a total of $r \times K$ action codewords, in the encoding process.

3. EXPERIMENTAL RESULTS

3.1. Datasets

We evaluate our proposed encoding scheme on two challenging action recognition benchmarks: HMDB51 [26] and UCF101 [12]. HMDB51 is comprised of 6,766 realistic action videos from 51 action classes. For evaluation, we use the available three train-test splits [26]. We perform performance evaluation using the average accuracy over these three train-test splits. UCF101 is a commonly used action recognition benchmark, comprised of 13,320 realistic video clips from 101 action classes. For evaluation purpose, we follow the provided three train-test splits and perform performance evaluation using the average accuracy on these splits.

3.2. Feature extraction

For feature extraction, we employ the Improved Dense Trajectories (iDT) based approach [6] to extract Histograms of HOG, HOF, MBHx and MBHy descriptors using the source code provided by the authors [6] with the default parameters. These descriptors are computed over the extracted trajectories and have dimensionality of 108 for HOF and 96 for HOG, MBHx and MBHy. iDT is the most popular state-of-the-art feature extraction approach, and it removes the invalid trajectories generated due to camera motion. Therefore, it is known as the improved version of [5]. We emphasise that our proposed DA-VLAD encoding can be used with any hand-crafted feature as well as with CNN based features.

3.3. Implementation details

For generating the action codewords from each class, we perform a class-specific k-means clustering (with $K = 256$) on 500K randomly selected feature descriptors (similar to [15]) from the training data. We choose $K = 256$ as it is standard codebook size for VLAD [14] and it is also considered as a best trade-off between computational complexity and accuracy [15]. Before applying the proposed feature encoding scheme, we first reduce the dimensionality of all of the four extracted features in half using Principal Component Analysis. In this way, the sizes of the feature descriptor become 54 for HOF and 48 for HOG, MBHx and MBHy. We apply the proposed DA-VLAD based encoding scheme to these four different feature descriptors separately, and we apply Power Normalization (PN) as done in [15]. Normalization is an essential step as we used different feature descriptors so as not to decrease the performance of the classification process negatively. For classification, we use a linear Support Vector Machine (SVM) in one-vs-all fashion with Cost=100.

3.4. Codewords ranking quality

In this experiment, we measure the quality of the proposed DA-VLAD encoding by seeing the effect of the discriminative importance of the action codewords. We measure the accuracy of the proposed DA-VLAD encoding on split 1 of the HMDB51 dataset by using the weights of the action codewords. We select the action codewords by ranking them depending upon their discriminative importance (i.e. their weights) using different threshold values λ. Fig. 3 shows the behavior of the action codewords by increasing the value of λ from 0 to 1. In Fig. 3, $\lambda \geq 0$ on the x-axis indicates that all action codewords are used in (2), similarly $\lambda \geq 0.1$ indicates that only action codewords with weights ≥ 0.1 are used.

From the results (Fig. 3), we observe a continuous boost in accuracy for all features and their combination by increasing λ from 0 to 1. The combination of four iDT features (HOG, HOF, MBHx and MBHy) is formed using early-fusion [18], i.e. by concatenating the DA-VLAD representations of all four features before classification. Using the combination of iDT features, action codewords with a weight equal to 1 ($\lambda = 1$) resulted in higher accuracy. From a total of $r \times K$ action codewords, we only select highly discriminative action codewords for DA-VLAD representation for all features which result in low dimensional representation of videos. From Fig. 4 it can be seen that a small portion of the action codewords are left using $\lambda = 1$ which reduces the corresponding DA-VLAD feature vector to 6%, 4%, 11% and 12% of the original $r \times K \times D$ length of HOG, HOF, MBHx and MBHy respectively. This analysis shows that using action codewords with weight equal to 1 are enough to represent a video efficiently. Therefore, we compute the average accuracy on three splits of the UCF101 and HMDB51 using the action codewords with weight equal to 1 with the early fusion of iDT features. As shown in Table 1, DA-VLAD achieves average accuracies of 95.1% and 80.1% on UCF101 and HMDB51 respec-

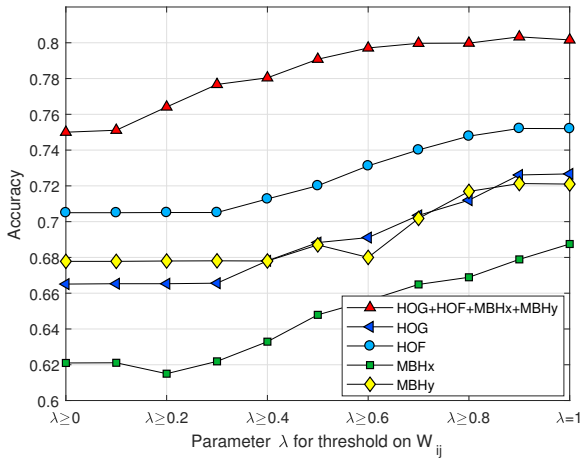

Fig. 3. Evaluation of action codeword ranking over split 1 of HMDB51 dataset.

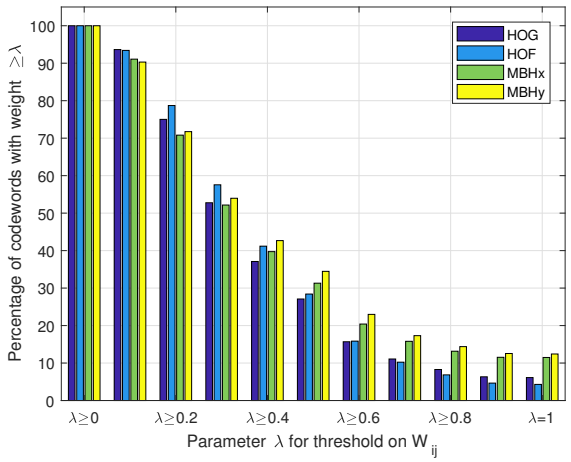

Fig. 4. Effect of ranking parameter λ on the number of action codewords for HMDB51 dataset

tively. Results (Table 1) show that DA-VLAD with weights equal to 1 provides more accuracy as compared to traditional VLAD when used with iDT features.

3.5. Comparison against state-of-the-art

Table 1 presents comparisons results of our proposal against the state-of-the-art approaches using average accuracy on the three splits of UCF101 and HMDB51 datasets. For this experiment, we use an early fusion of iDT features, and we select the action codewords with weights equal to 1 for DA-VLAD encoding. Results show that our encoding scheme achieves improved accuracies compared with the state-of-the-art approaches by a fairly large margin. In these experiments the increases are 7.0% on the HMDB51, and 0.8% on the UCF101 compared to the leading approach [15] which uses deep features in combination with iDT and Histograms of motion gradients (HMG) [27]. One can observe more improvement in accuracy for HMDB51 than UCF101 dataset. This is because UCF101 is more complex with a large number of action classes than HMDB51 dataset. DA-VLAD encoding, using simple iDT features, outperforms other methods (Long-term temporal convolutions (LTC) [9], [8, 10, 17–20]) which reported their results using iDT features in combination with CNN features. Our method got about 21% increase in accuracy as compared to Modified VLAD [21] that uses hand-crafted features (HOF descriptor).

Table 1. Comparison of DA-VLAD with different encoding methods on UCF101 and HMDB51 averaged over 3 splits

Methods	UCF101	HMDB51
iDT+VLAD [28]	73.1	52.1
HOF+Modified VLAD [21]	74.1	-
DT+MVSV [23]	83.5	55.9
iDT+iFV [6]	85.9	57.2
iDT+Hybrid [22]	87.9	61.1
iDT+MoFAP [19]	88.3	61.7
iDT+C3D [10]	90.4	-
iDT+C3D AdaScan [18]	93.2	66.9
iDT+GRP [20]	92.3	67
iDT+LTC [9]	92.7	67.2
iDT+ST-VLAD [16]	91.5	67.6
iDT+Two-Stream Fusion [8]	93.5	69.2
iDT+ActionVLAD(VGG-16) [17]	93.6	69.8
iDT+ST-VLMPF [15]	94.3	73.1
Our: iDT+DA-VLAD	**95.1**	**80.1**

4. CONCLUSION

We have proposed a novel encoding scheme based on the discriminative power of action codewords, called DA-VLAD, for recognizing human actions in videos. DA-VLAD is formulated on the observation that the human action videos contain a large number of non-informative and overlapping feature points which are common in different action classes resulting in a poorer representation of videos. DA-VLAD exploited the discriminative power of action codewords in the form of weights which define the contribution of the action codewords in the encoding process. Through experiments, we concluded that using only action codewords with weight equal to 1 have resulted in higher accuracy rate than using all action codewords. DA-VLAD outperformed the state-of-the-art approaches using iDT features on HMDB51 and UCF101. In future, we can use DA-VLAD with CNN features, and we can also test our proposed weighted action codeword scheme with iFV and other encoding schemes.

5. ACKNOWLEDGMENTS

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research. We also acknowledge the support from the Directorate of Advance Studies, Research and Technological development (ASR) & TD, University of Engineering and Technology Taxila, Pakistan. Sergio A Velastin acknowledges funding by the Universidad Carlos III de Madrid, the European Unions Seventh Framework Programme for research, technological development and demonstration under grant agreement n 600371, el Ministerio de Economia y Competitividad (COFUND2013-51509) and Banco Santander.

6. REFERENCES

[1] Navneet Dalal and Bill Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. IEEE, 2005, vol. 1, pp. 886–893.

[2] Ivan Laptev, Marcin Marszalek, Cordelia Schmid, and Benjamin Rozenfeld, "Learning realistic human actions from movies," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[3] Navneet Dalal, Bill Triggs, and Cordelia Schmid, "Human detection using oriented histograms of flow and appearance," in *European conference on computer vision*. Springer, 2006, pp. 428–441.

[4] Jasper Uijlings, Ionut C Duta, Enver Sangineto, and Nicu Sebe, "Video classification with densely extracted hog/hof/mbh features: an evaluation of the accuracy/computational efficiency trade-off," *International Journal of Multimedia Information Retrieval*, vol. 4, no. 1, pp. 33–44, 2015.

[5] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu, "Dense trajectories and motion boundary descriptors for action recognition," *International journal of computer vision*, vol. 103, no. 1, pp. 60–79, 2013.

[6] Heng Wang and Cordelia Schmid, "Action recognition with improved trajectories," in *Proceedings of the IEEE international conference on computer vision*, 2013, pp. 3551–3558.

[7] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning.," in *AAAI*, 2017, pp. 4278–4284.

[8] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman, "Convolutional two-stream network fusion for video action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1933–1941.

[9] Gul Varol, Ivan Laptev, and Cordelia Schmid, "Long-term temporal convolutions for action recognition," *IEEE transactions on pattern analysis and machine intelligence*, 2017.

[10] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497.

[11] Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta, "Hollywood in homes: Crowdsourcing data collection for activity understanding," in *European Conference on Computer Vision*. Springer, 2016, pp. 510–526.

[12] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *arXiv preprint arXiv:1212.0402*, 2012.

[13] Florent Perronnin, Jorge Sánchez, and Thomas Mensink, "Improving the fisher kernel for large-scale image classification," *Computer Vision–ECCV 2010*, pp. 143–156, 2010.

[14] Herve Jegou, Florent Perronnin, Matthijs Douze, Jorge Sánchez, Patrick Perez, and Cordelia Schmid, "Aggregating local image descriptors into compact codes," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 9, pp. 1704–1716, 2012.

[15] Ionut Cosmin Duta, Bogdan Ionescu, Kiyoharu Aizawa, and Nicu Sebe, "Spatio-temporal vector of locally max pooled features for action recognition in videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3097–3106.

[16] Ionut C Duta, Bogdan Ionescu, Kiyoharu Aizawa, and Nicu Sebe, "Spatio-temporal vlad encoding for human action recognition in videos," in *International Conference on Multimedia Modeling*. Springer, 2017, pp. 365–378.

[17] Rohit Girdhar, Deva Ramanan, Abhinav Gupta, Josef Sivic, and Bryan Russell, "Actionvlad: Learning spatio-temporal aggregation for action classification," *arXiv preprint arXiv:1704.02895*, 2017.

[18] Amlan Kar, Nishant Rai, Karan Sikka, and Gaurav Sharma, "Adascan: Adaptive scan pooling in deep convolutional neural networks for human action recognition in videos," *arXiv preprint arXiv:1611.08240*, 2016.

[19] Limin Wang, Yu Qiao, and Xiaoou Tang, "Mofap: A multi-level representation for action recognition," *International Journal of Computer Vision*, vol. 119, no. 3, pp. 254–271, 2016.

[20] Anoop Cherian, Basura Fernando, Mehrtash Harandi, and Stephen Gould, "Generalized rank pooling for activity recognition," *arXiv preprint arXiv:1704.02112*, 2017.

[21] Ionuţ Mironică, Ionuţ Cosmin Duţă, Bogdan Ionescu, and Nicu Sebe, "A modified vector of locally aggregated descriptors approach for fast video classification," *Multimedia Tools and Applications*, vol. 75, no. 15, pp. 9045–9072, 2016.

[22] Xiaojiang Peng, Limin Wang, Xingxing Wang, and Yu Qiao, "Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice," *Computer Vision and Image Understanding*, vol. 150, pp. 109–125, 2016.

[23] Zhuowei Cai, Limin Wang, Xiaojiang Peng, and Yu Qiao, "Multi-view super vector for action recognition," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2014, pp. 596–603.

[24] Hui Lv, Tao Lei, Xianglin Huang, and Yakun Zhang, "Dual adaptive representation of vector of locally aggregated," in *Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), 2015 IEEE*. IEEE, 2015, pp. 686–689.

[25] David Picard and Philippe-Henri Gosselin, "Improving image similarity with vectors of locally aggregated tensors," in *Image Processing (ICIP), 2011 18th IEEE International Conference on*. IEEE, 2011, pp. 669–672.

[26] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre, "Hmdb: a large video database for human motion recognition," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2556–2563.

[27] Ionut C Duta, Jasper RR Uijlings, Tuan A Nguyen, Kiyoharu Aizawa, Alexander G Hauptmann, Bogdan Ionescu, and Nicu Sebe, "Histograms of motion gradients for real-time video classification," in *Content-Based Multimedia Indexing (CBMI), 2016 14th International Workshop on*. IEEE, 2016, pp. 1–6.

[28] Mihir Jain, Herve Jegou, and Patrick Bouthemy, "Better exploiting motion for better action recognition," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013, pp. 2555–2562.